PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Robert A. Relyea et al.	Examiner:	Insun Kang
Serial No.:	10/715,709	Group Art Unit:	2193
Filed:	November 18, 2003	Docket No.:	60001.318US01
Title:	METHOD AND SYSTEM FOR MAPPING TAGS TO CLASSES USING NAMESPACES		

ELECTRONICALLY FILED ON OCTOBER 29, 2008

REQUEST FOR CONTINUED EXAMINATION UNDER 37 C.F.R. §1.114

Mail Stop RCE
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

27488
PATENT TRADEMARK OFFICE

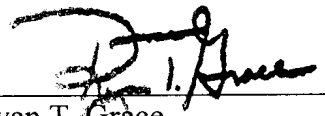
Dear Sir:

Under 37 C.F.R. §1.114, it is respectfully requested that this application be accorded the benefits of Continued Examination. A copy of the Amendment filed on June 27, 2008, is attached hereto.

The amount of $810 of which is to cover the required fee for a large entity is being paid via credit card.

Respectfully submitted,

MERCHANT & GOULD P.C.
P.O. Box 2903
Minneapolis, MN 55402-0903
612-332-5300



Ryan T. Grace
Reg. No. 52,956

RESPONSE UNDER 37 C.F.R. 1.116
EXPEDITED PROCEDURE
EXAMINING GROUP 2193

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Robert A. Relyea et al.	Examiner:	Insun Kang
Serial No.:	10/715,709	Group Art Unit:	2193
Filed:	November 18, 2003	Docket No.:	60001.318US01
Title:	METHOD AND SYSTEM FOR MAPPING TAGS TO CLASSES USING NAMESPACES		

ELECTRONICALLY FILED JUNE 27, 2008

AMENDMENT UNDER 37 C.F.R. § 1.116

Mail Stop AF
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Dear Examiner:

In response to the Office Action of April 30, 2008, please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks/Arguments begin on page 7 of this paper.

Amendments to the Claims:

1 (Previously Presented): A method for mapping a tag in a markup language (ML) document to a class using namespaces, comprising:

analyzing a tag in the ML document;

referencing a definition file location attribute in the ML document, wherein the definition file location attribute is identified by the tag;

retrieving a definition file from a location identified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace;

referencing a common language runtime namespace related to the tag within the definition file to determine the common language runtime class associated with the tag; and

locating the common language runtime class in an assembly such that the tag is mapped to the common language runtime class.

2 (Original): The method of Claim 1, wherein analyzing a tag further comprises analyzing the tags in linear order as listed in the ML document.

3 (Previously Presented): The method of Claim 1, wherein analyzing a tag further comprises reading a prefix corresponding to an ML namespace related to the tag.

4 (Previously Presented): The method of Claim 3, further comprising defining the ML namespace using the prefix, wherein the prefix maps to an extensible markup language namespace, and wherein the definition file maps the extensible markup language namespace to a common language runtime namespace and the assembly.

5 (Original): The method of Claim 3, wherein the prefix is defined in the ML document.

6 (Original): The method of Claim 1, further comprising determining whether the definition file is available locally in a cache, and if not available, storing the retrieved definition file in the cache.

7 (Original): The method of Claim 1, wherein retrieving a definition file further comprises retrieving the definition file from a network location specified by definition file location attribute.

8 (Previously Presented): The method of Claim 1, wherein locating the common language runtime class in an assembly further comprises locating the common language runtime class in a dynamic link library, the dynamic link library comprising common language runtime classes of functions associated with the common language runtime namespace of the definition file.

9 (Original): The method of Claim 1, further comprising generating the ML document, the ML document comprising the tag and the definition file location attribute.

10 (Previously Presented): The method of Claim 1, wherein the definition file comprises a list of the common language runtime namespaces, schemas and assemblies associated with the common language runtime class related to the common language runtime namespace.

11 (Original): The method of Claim 1, wherein the namespace of the definition file is associated with a property within an element of the ML document.

12 (Previously Presented): A computer-readable storage medium having computer-executable instructions for mapping a tag in an ML document to a common language runtime class using common language runtime namespaces, the instructions comprising:

evaluating a tag in the ML document, wherein evaluating the tag comprises reading a prefix associated with an ML namespace when the prefix is present;

detecting a definition file location attribute associated with the tag in the ML document;

fetching a definition file from a location specified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace;

resolving the common language runtime namespace related to the tag within the definition file to establish the common language runtime class associated with the tag; and

finding an assembly that includes the common language runtime class such that the tag is mapped to the common language runtime class, wherein the assembly comprises common language runtime classes of functions associated with the common language runtime namespace.

13 (Previously Presented): The computer-readable storage medium of Claim 12, further comprising determining whether the definition file is available locally in a cache, and if not available, storing the fetched definition file in the cache.

14 (Previously Presented): The computer-readable storage medium of Claim 12, wherein the definition file is fetched from a network location.

15 (Previously Presented): The computer-readable storage medium of Claim 12, further comprising defining the ML namespace using the prefix, wherein the prefix maps to an extensible markup language namespace, and wherein the definition file maps the extensible markup language namespace to a common language runtime namespace and the assembly.

16 (Previously Presented): The computer-readable storage medium of Claim 12, wherein the assembly comprises a dynamic link library.

17 (Previously Presented): The computer-readable storage medium of Claim 12, wherein the definition file comprises a list of the common language runtime namespaces, schemas and assemblies associated with the common language runtime class related to the common language runtime namespace.

18 (Previously Presented): The computer-readable storage medium of Claim 12, wherein the common language runtime namespace of the definition file is associated with a property within an element of the ML document.

19 (Currently Amended): A system for mapping a tag in an ML document to a common language runtime class using common language runtime namespaces, the system comprising ~~comprises~~:

a processor; and

a memory having computer-executable instructions, the computer-executable instructions being configured for:

~~means for~~ analyzing a tag in the ML document;

~~means for~~ referencing a definition file location attribute in the ML document, wherein the definition file location attribute is related to the tag;

~~means for~~ retrieving a definition file from a location specified by the definition file location attribute, wherein the definition file includes:

a schema that limits the scope of attributes in the definition file,

a list of assemblies that reference the definition file,

a list of common language runtime namespaces associated with the list of assemblies that reference the definition file, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace, and

an installation tag that includes a uniform resource identifier for installing assemblies of the list of assemblies;

~~a list of common language runtime namespaces, wherein each common~~

~~language runtime namespace includes a list of common language classes associated with the common language runtime namespace;~~

~~means for~~ referencing a common language runtime namespace related to the tag within the definition file to determine the common language runtime class associated with the tag; and

~~means for~~ locating the common language runtime class in an assembly of the list of assemblies such that the tag is mapped to the common language runtime class.

20 (Currently Amended): The system of Claim 19, wherein ~~the means for~~ analyzing the tag includes reading ~~reads~~ a prefix associated with the ~~an~~ ML namespace when the prefix is present, wherein the prefix maps to an extensible markup language namespace, and wherein the definition file maps the extensible markup language namespace to a common language runtime namespace and the assembly.

REMARKS/ARGUMENTS

With regard to claims 1-18, they have not been further amended. Applicants believe that current claims 1-18 are allowable over the cited references as more fully described below. Independent claim 19 has been amended as set forth above. Dependent claim 20 has been amended as set forth above to correspond to the changes made in independent claim 19. With regard to claim 4, in the last response, claim 4 included the identifier "original". This was a typographical error as indicated above. Claim 4 now includes the identifier "previously presented."

I. Examiner Interview

Applicants requested an interview on June 2. The interview request was not accepted. An interview was not conducted.

II. Rejection Under 35 U.S.C. § 103(a)

Claims 1-5, 7-12, and 14-20 are rejected under 35 U.S.C. § 103(a) as being unpatentable over "XML Namespaces by Example, 1999" published to Bray (hereinafter "Bray") in view of "What's in a namespace," techrepublic.com, CNET networks, Inc., published on 5/29/2002 to Lurie et al. (hereinafter "Lurie"). Claims 6 and 13 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Bray in view of Lurie and further in view of U.S. Publication No. 2004/0103199 published to Chao et al. (hereinafter "Chao"). Applicants respectfully disagree with the rejections. Independent claim 1 includes the following combination of features that is not taught or otherwise suggested by the cited references:

> analyzing a tag in the ML document;
>
> referencing a definition file location attribute in the ML document, wherein the definition file location attribute is identified by the tag;
>
> ***retrieving a definition file from a location identified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the***

common language runtime namespace;

referencing a common language runtime namespace related to the tag within the definition file to determine the common language runtime class associated with the tag; and

locating the common language runtime class in an assembly such that the tag is mapped to the common language runtime class.

Contrary to the assertions in the Office Action, the references do not teach or otherwise suggest the above combination of features. On page 1, Bray teaches an example of XML, in part, as follows:

<h: html xmlns: xdc="http: //www.xml.com/books"
xmlns: h=http://www.w3.org/HTML/1998/html4

Bray continues by teaching as follows:

In this example, the elements prefixed with xdc are associated with a namespace whose name is http: //www.xml.com/books, while those prefixed with h are associated with a namespace whose name is http://www.w3.org/HTML/1998/html4. (Bray, at page 1)

Bray further teaches as follows:

The only purpose of the namespaces is to give programmers a helping hand, enabling them to process the tags and attributes they care about and ignore those that don't matter to them.... The only reason namespaces exist, once again, is to give elements and attributes programmer-friendly names that will be unique across the whole Internet. (Bray, at page 3)

Here, Bray is teaching that http: //www.xml.com/books and http://www.w3.org/HTML/1998/html4 are the names of the namespace. Prefixes are used to associate with the namespaces above. Bray continues by teaching that **the only purpose** for the namespace is to provide programmer-friendly names. The current specification, however, teaches using a prefix to identify a definition file location attribute in the ML document. The definition file is then obtained to map the tag in the ML document to a common language runtime class. Claim 1 recites "retrieving a definition file from a location identified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of

common language classes associated with the common language runtime namespace," "referencing a common language runtime namespace related to the tag within the definition file to determine the common language runtime class associated with the tag," in combination with "locating the common language runtime class in an assembly such that the tag is mapped to the common language runtime class." Again, Bray is teaching that the http: //www.xml.com/books and http://www.w3.org/HTML/1998/html4 **are the names of the namespace**. Claim 1 makes clear that a definition file is referenced in the ML and obtained. The **definition file** is then used to map the **tag** to a common language runtime **class**. Bray does not mention associating a definition file with the ML document, obtaining a definition file, or mapping the tag to a class based on the definition file. The reference in Bray is the namespace itself, **not a definition file**.

With regard to Lurie, Lurie does not remedy the lack of teaching in Bray. FIGURE A in Lurie shows how classes are divided up in the namespaces that compose the .NET CLR. (Lurie, page 5). Lurie fails to teach "retrieving a definition file from a location identified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace," "referencing a common language runtime namespace related to the tag within the definition file to determine the common language runtime class associated with the tag," in combination with "locating the common language runtime class in an assembly such that the tag is mapped to the common language runtime class." Again, Lurie does not teach or suggest that a definition file is referenced in the ML and obtained. The **definition file** is then used to map the **tag** to a common language runtime **class**. Lurie does not mention associating a definition file with the ML document, obtaining a definition file, or mapping the tag to a class based on the definition file. Accordingly, applicants assert that claim 1 is allowable over the cited references.

Independent claim 12 includes the following combination of features that is not taught or otherwise suggested by the cited references:

> evaluating a tag in the ML document, wherein evaluating the tag comprises reading a prefix associated with an ML namespace when the prefix is present;

detecting a definition file location attribute associated with the tag in the ML document;

fetching a definition file from a location specified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace;

resolving the common language runtime namespace related to the tag within the definition file to establish the common language runtime class associated with the tag; and

finding an assembly that includes the common language runtime class such that the tag is mapped to the common language runtime class, wherein the assembly comprises common language runtime classes of functions associated with the common language runtime namespace.

Contrary to the assertions in the Office Action, the references do not teach or otherwise suggest the above combination of features. On page 1, Bray teaches an example of XML, in part, as follows:

<h: html xmlns: xdc="http: //www.xml.com/books"
xmlns: h=http://www.w3.org/HTML/1998/html4

Bray continues by teaching as follows:

In this example, the elements prefixed with xdc are associated with a namespace whose name is http: //www.xml.com/books, while those prefixed with h are associated with a namespace whose name is http://www.w3.org/HTML/1998/html4. (Bray, at page 1)

Bray further teaches as follows:

The only purpose of the namespaces is to give programmers a helping hand, enabling them to process the tags and attributes they care about and ignore those that don't matter to them.... The only reason namespaces exist, once again, is to give elements and attributes programmer-friendly names that will be unique across the whole Internet. (Bray, at page 3)

Here, Bray is teaching that http: //www.xml.com/books and http://www.w3.org/HTML/1998/html4 are the names of the namespace. Prefixes are used to

associate with the namespaces above. Bray continues by teaching that **the only purpose** for the namespace is to provide programmer-friendly names. The current specification, however, teaches using a prefix to identify a definition file location attribute in the ML document. The definition file is then obtained to map the tag in the ML document to a common language runtime class. Claim 12 recites "fetching a definition file from a location specified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace," "resolving the common language runtime namespace related to the tag within the definition file to establish the common language runtime class associated with the tag," in combination with "finding an assembly that includes the common language runtime class such that the tag is mapped to the common language runtime class, wherein the assembly comprises common language runtime classes of functions associated with the common language runtime namespace." Again, Bray is teaching that the http: //www.xml.com/books and http://www.w3.org/HTML/1998/html4 **are the names of the namespace**. Claim 12 makes clear that a definition file is referenced in the ML and obtained. The **definition file** is then used to map the **tag** to a common language runtime **class**. Bray does not mention associating a definition file with the ML document, obtaining a definition file, or mapping the tag to a class based on the definition file. The reference in Bray is the namespace itself **not a definition file**.

With regard to Lurie, Lurie does not remedy the lack of teaching in Bray. FIGURE A in Lurie shows how classes are divided up in the namespaces that compose the .NET CLR. (Lurie, page 5). Lurie fails to teach "fetching a definition file from a location specified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace," "resolving the common language runtime namespace related to the tag within the definition file to establish the common language runtime class associated with the tag," in combination with "finding an assembly that includes the common language runtime class such that the tag is mapped to the common language runtime class, wherein the assembly comprises common language runtime

classes of functions associated with the common language runtime namespace." Again, Lurie does not teach or suggest that a definition file is referenced in the ML and obtained. The **definition file** is then used to map the **tag** to a common language runtime **class**. Lurie does not mention associating a definition file with the ML document, obtaining a definition file, or mapping the tag to a class based on the definition file. Accordingly, applicants assert that claim 12 is allowable over the cited references.

Independent claim 19 has been amended to clarify the following combination of features that is not taught or otherwise suggested by the cited references:

> a processor; and
>
> a memory having computer-executable instructions, the computer-executable instructions being configured for:
>
> analyzing a tag in the ML document;
>
> referencing a definition file location attribute in the ML document, wherein the definition file location attribute is related to the tag;
>
> ***retrieving a definition file from a location specified by the definition file location attribute, wherein the definition file includes:***
>
> ***a schema that limits the scope of attributes in the definition file,***
>
> ***a list of assemblies that reference the definition file,***
>
> ***a list of common language runtime namespaces associated with the list of assemblies that reference the definition file, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace, and***
>
> ***an installation tag that includes a uniform resource identifier for installing assemblies of the list of assemblies;***
>
> ***referencing a common language runtime namespace related to the tag within the definition file to determine the common language runtime class associated with the tag; and***

locating the common language runtime class in an assembly of the list of assemblies such that the tag is mapped to the common language runtime class.

Contrary to the assertions in the Office Action, the references do not teach or otherwise suggest the above combination of features. With regard to Bray and Lurie, applicants rely on the arguments set forth above. Furthermore, independent claim 19 has been amended to clarify that the definition file includes "a schema that limits the scope of attributes in the definition file." Independent claim 19 has been further amended to clarify that the definition file includes "a list of assemblies that reference the definition file." Also, independent claim 19 has been amended to recite that the definition file includes "a list of common language runtime namespaces associated with the list of assemblies that reference the definition file, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace." Furthermore, claim 19 has been amended to clarify that the definition file includes "an installation tag that includes a uniform resource identifier for installing assemblies of the list of assemblies." Applicants assert that the clarifications associated with independent claim 19 are not taught or otherwise suggested by the cited references. Accordingly, applicants believe that independent claim 19 is allowable over the cited references.

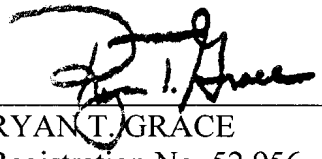
With regard to the dependent claims, they include features that are not taught or suggested by the cited references. Furthermore, the dependent claims ultimately depend from the independent claims, respectively. As such, they should be found allowable for at least those same reasons.

III. Request for Reconsideration

In view of the foregoing amendments and remarks, all pending claims are believed to be allowable and the application is in condition for allowance. Therefore, a Notice of Allowance is respectfully requested. Should the Examiner have any further issues regarding this application, the Examiner is requested to contact the undersigned attorney for the applicant at the telephone number provided below.

Respectfully submitted,

MERCHANT & GOULD P.C.



RYAN T. GRACE
Registration No. 52,956
Direct Dial: 402.344.3000

MERCHANT & GOULD P.C.
P. O. Box 2903
Minneapolis, Minnesota 55402-0903
206.342.6200